Facial expression recognition by combination of classifiers

Séverine Dubuisson and Franck Davoine
Laboratory Heudiasyc, University of Technology of Compiègne
BP 20529. F-60205 Compiègne, France
Tel: +33 3 44 23 44 82; fax: +33 3 44 23 44 77
e-mail: sdubuiss@hds.utc.fr

ABSTRACT

In this paper, we present a classifier fusion solution for automatic facial expression recognition. We represent our data using a sorted Principal Component Analysis, followed by a Linear Discriminant Analysis: the selection of principal components first performs a dimensionally reduction by improving discriminant capacities and then, a Linear Discriminant Analysis provides a class representation subspace where new samples can be classified. Using a fuzzy integral method [7], the classification is operated by combining, the outputs of three classifiers (using Mahalanobis distance, Euclidean distance and a Bayes rule based criterion). This method gives, for a new sample, a probabilistic interpretation of the different classifier outputs to generate a fuzzy measure vector for each considered facial expression class. The sample is then classified into class with maximum fuzzy posterior probability.

1 INTRODUCTION

Face analysis can be divided into several subgroups: face detection, facial feature extraction, and recognition (such as face, age, gender, race, pose and facial expression recognition). The problem of facial expression recognition has recently emerged. In a general way, an expression is a combination of the Action Units (AUs) defined by Ekman and Friesen [5]: a local facial motion resulting from the compression or relaxation of facial muscles. A facial expression can be seen in two different ways: a motion in the face which requires working with video sequences and face motion analysis tools, or the shape and texture of the face, using statistical analysis tools, local filtering or facial feature measurement. The facial motion has been the first way to be explored by the researchers: a facial expression can be described by a global facial motion model [1], or by a set of local facial motion models [12] that researchers analyze along a video sequence by tracking facial feature points. In statistical analysis, we first have to learn what we are looking for: a learning method of the appearance of facial expressions using an eigenvector decomposition of the image space has been proposed in [10], which build a new representation subspace where a likelihood measure is computed to analyze new facial expression images. Local filtering methods change the domain of the images, which can help in feature extraction. Gabor wavelet filters showed to perform well for facial expression analysis [9, 3] because they remove most of the variability in images: they have been found to be particularly suitable for image decomposition and representation when the goal is the derivation of local and discriminant features.

2 DATA REPRESENTATION

Multivariate statistical analysis define general tools for describing, compressing and analyzing data: they allow to derive a statistical model from a learning set by projecting its samples onto a lower dimensional representation subspace. We first perform Principal Component Analysis (PCA) to reduce the dimensionality of the input data, and then we sort out the principal components into the order of their importance for the facial expression problem. Finally, Linear Discriminant Analysis, applied into this sorted eigenspace, provides a discriminant representation subspace. Both techniques require precise normalization and registration of faces, that are described next section.

2.1 Data extraction and normalization

Most of a facial expression information is concentrated around facial features such as eyes or mouth: including irrelevant parts (hair, background, ...) can generate incorrect decisions for expression recognition. We perform a facial mask extraction by manually positioning four facial feature points: the pupil centers, the top of the nose and the middle of the mouth. Two affine transformations, applied independently on the top and bottom parts of faces, are used in such a way that these four points are located in fixed positions in target images. We then crop right and left lower lateral parts of faces to only consider their internal area (face shape can be an information which perturb the recognition process). Some examples are given in Figure 1.

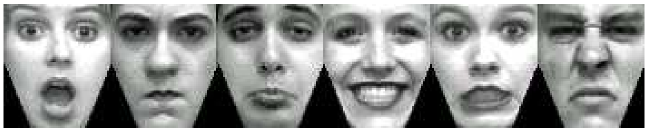

Figure 1: Manually extracted facial masks for different facial expressions (CMU-Pittsburgh database [6]).

2.2 Sorted Principal Component Analysis

PCA is an unsupervised linear feature extraction method which has been proposed for face representation, identification, recognition or detection in [10]. Let $S = \{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$ be the centered learning set containing N d-dimensional facial mask vectors $\mathbf{x}_i$, and $C = SS^T$ its covariance matrix. PCA [11] seeks the linear transformation matrix W that maps the original space onto a N-dimensional subspace, with $N \ll d$, by factorizing the covariance matrix into the following form:

$$C = W \Lambda W^T$$

where W is an orthonormal nonzero eigenvector matrix and Λ a diagonal eigenvalue matrix with diagonal elements sorted out in decreasing order ($\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$). The most expressive vectors derived from PCA are those corresponding to the leading largest eigenvalues: N principal axes (eigenfaces) are used to derive N-dimensional feature vector $\mathbf{y}$ for each d-dimensional sample $\mathbf{x}$ so that: $\mathbf{y} = W^T \mathbf{x}$.

Selection of the principal components

We propose to make Principal Component Analysis suitable for a facial expression recognition task, by selecting principal components, in order to construct a low-dimensional discriminant eigenspace. Such analysis will be called, from now, a *Sorted PCA*. PCA returns a set of N principal components among which we are seeking the K most discriminant, that we call the "optimal" ones. For this, we consider an iterative supervised feature selection process, described in [4], that progressively selects principal components to construct an optimal projection base, by applying the following "step by step" method:

- During step 1, we seek the optimal principal component among the N available.
- During step j, we seek the principal component (among the $N - j + 1$ remaining) which, when added to those previously kept, is the new optimal one.

The selection of an optimal principal component is achieved by maximizing the general class separability measure, defined by the Fisher criterion F, which can be expressed as:

$$F = \max_{Comp} \frac{|S_B|}{|S_W|}$$

where $Comp$ is the set of principal components, S_W and S_B are respectively the within- and between-class scatter matrix. Once all the principal components have been sorted, the final dimension K of the optimal subspace corresponds to the minimum of the generalization error rate profile, e.g we seek the rank K for which the addition of a new component to the optimal set does not decrease this error anymore.

2.3 Linear Discriminant Analysis

These projected M-dimensional samples are then used to execute LDA: we determine the mapping which simultaneously maximizes the between-class scatter S_B while minimizing within-class scatter S_W of the projected samples, so that the classes are separated as best as possible. The way to find the required mapping is to maximize the quantity trace($S_W^{-1} S_B$). This is done by solving the eigenvalue problem [11]:

$$S_W^{-1} S_B W = W \Lambda$$

Column vectors of matrix W correspond to the Fisherfaces [2], and the final representation subspace dimension for a c class problem is $(c - 1)$.

2.4 Representation subspace

We use CMU-Pittsburgh database images [6] to construct a learning set containing $N = 210$ facial masks (size 60×70: $d = 4200$-pixel vectors): $N_c = 35$ per expression and the $c = 6$ universal facial expressions (surprise, sadness, fear, anger, disgust and joy). We first perform a PCA, then we sort the principal components to provide a 55-dimensional subspace, and finally we apply a LDA to construct the 5-dimensional discriminant subspace. The projection of this set onto the Sorted PCA + LDA subspace is shown on Figure 2.

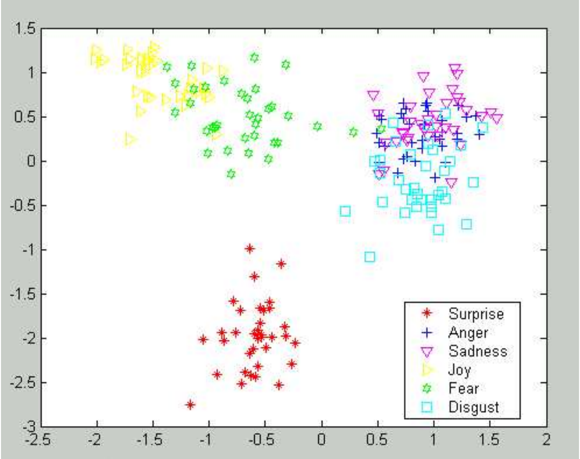

Figure 2: Projection of the learning set onto the subspace constructed by Sorted PCA plus LDA.

3 DATA CLASSIFICATION

A statistical analysis of training data allows to generate a subspace in which a new sample can be projected and then classified. The usual method consists in determining a degree of similarity between this sample and each of the classes. This can be done by measuring a statistical distance (Euclidean and Mahalanobis) or by applying classification rules (k nearest-neighbors), to determine the geometrical proximity between the sample and each class. Bayesian rule based classification also estimates the posterior probability of each class for a given sample. Each of these classifiers can supply different but useful results: we then have chosen to combine the outputs of some of them to classify new samples.

A classical method consists in multiplying between them the ith ($i = 1, \cdots, c$, where c is the number of classes of the problem: i identify a class) output of each classifiers. Unfortunately, we should make the hypothesis that the classifiers are mutually independents. We also can associate the sample with the class which posterior probability estimation is the greatest, for every merged classifiers. This solution is not satisfactory, because the outputs of different classifiers are not comparable. So, we make the choice to apply a fuzzy integral method, initially proposed in [7], which is described in next section,

3.1 Principle

The fuzzy integral method makes, for a tested sample, a fuzzy aggregation of the classifier outputs. If we consider L different classifiers, for a c-class problem, we define the decision profile matrix DP as:

$$DP = \begin{pmatrix} [\mu_1^1(\mathbf{x}) & \mu_2^1(\mathbf{x}) & \cdots & \mu_c^1(\mathbf{x})] \\ [\mu_1^2(\mathbf{x}) & \mu_2^2(\mathbf{x}) & \cdots & \mu_c^2(\mathbf{x})] \\ \vdots & \vdots & \cdots & \vdots \\ [\mu_1^L(\mathbf{x}) & \mu_2^L(\mathbf{x}) & \cdots & \mu_c^L(\mathbf{x})] \end{pmatrix}$$

where $\mu_i^j(\mathbf{x})$ is, for a given sample $\mathbf{x}$, to the posterior probability, estimated by the jth classifier, of the ith class: each column i ($i = 1, \cdots, c$) corresponds to a vector containing L fuzzy measure values for the ith class.

3.2 Fuzzy integral method

Fuzzy integration is interpreted as searching for the maximal grade of agreement between the sorted classifier outputs for class i (objective evidence) and the expectation, given by L fussy measure values. The problem is then to find a fuzzy measure vector for each class. To find this support vector for class i, $\mu_i^{\tilde{D}}$, we apply the following algorithm [8]:

1. Fix L fuzzy densities g^i, $i = 1, \cdots, L$, such as $g^i = \frac{1}{L}$.
2. Compute $\lambda > 1$ as the only real root greater than -1 of the equation:
$$\lambda + 1 = \prod_{i=1}^{L}(1 + \lambda g^i)$$
3. For a given $\mathbf{x}$, sort the kth column of $DP(\mathbf{x})$ in decreasing order to obtain $[d_{i_1,k}(\mathbf{x}), \cdots, d_{i_L,k}(\mathbf{x})]$, with $d_{i_j,k}(\mathbf{x}) > d_{i_{j+1},k}(\mathbf{x})$.
4. Sort, in the same way, the corresponding fuzzy densities $g^{i_1}, \cdots, g^{i_L}$.
5. Set $g(1) = g^{i_1}$.
6. For $t = 2$ to L: $g(t) = g^{i_t} + (1 + \lambda g^{i_t})g(t-1)$
7. Compute the degree of support for class k as:
$$\mu_i^{\tilde{D}} = \max_{t=1}^{L}\{\min\{d_{i_t,k}, g(t)\}\}$$

For a specific new sample $\mathbf{x}$, we get a fuzzy measure different for each class, corresponding to a posterior fuzzy probability estimation.

3.3 Application

In our case, we consider a $c = 6$-class problem and use three different classifiers ($L = 3$), using Euclidean distance, Mahalanobis distance and the Bayes rule based criterion (see [11] for details concerning these classifiers). Each of the classifier outputs have to be written as a vector $\mathbf{w}$ such as:

$$\mathbf{w} = [\mu_1(\mathbf{x}), \cdots, \mu_c(\mathbf{x})]$$

where $\mu_i(\mathbf{x})$ is the posterior probability estimation of the ith class for the sample $\mathbf{x}$.

The Euclidean D^2_{euc} and Mahalanobis D^2_{mahal} distances between two samples $\mathbf{x}$ and $\mathbf{y}$ are given by the formula:

$$D^2_{\text{euc}} = (\mathbf{x} - \mathbf{y})^t(\mathbf{x} - \mathbf{y})$$
$$D^2_{\text{mahal}} = (\mathbf{x} - \mathbf{y})^t\Sigma^{-1}(\mathbf{x} - \mathbf{y})$$

where Σ is the total scatter matrix of the data.

Mahalanobis and Euclidean distance based classifier outputs are vectors $\mathbf{w}_d = [d_1(\mathbf{x}), \cdots, d_c(\mathbf{x})]$, containing geometrical distances, which do not correspond to probability values. We then have to transform these outputs so that:

$$\mathbf{w}_d = [d_1(\mathbf{x}), \cdots, d_c(\mathbf{x})] \longmapsto \mathbf{w} = [\mu_1(\mathbf{x}), \cdots, \mu_c(\mathbf{x})]$$
$$\text{with} \quad \mu_i(\mathbf{x}) = \frac{d_i(\mathbf{x})}{\sum_{j=1}^{c} d_j(\mathbf{x})}$$

The Bayes rule is based on the minimization of the probability error, for the cost $\{0, 1\}$. It associates a sample

$\mathbf{x}$ with the class ω_i if its posterior probability estimation $P(\omega_i|\mathbf{x})$ is maximum:

$$\mathbf{x} \to \omega_i \qquad \text{if} \qquad p(\omega_i)f(\mathbf{x}|\omega_i) > p(\omega_j)f(\mathbf{x}|\omega_j) \;\; \forall j \neq i$$

If we make the Gaussian hypothesis, the Bayes rule consists in finding the class ω_i which maximizes:

$$g_i(\mathbf{x}) = \ln p(\omega_i) - \frac{1}{2}\ln|\Sigma_i| - \frac{1}{2}(\mathbf{x} - \tilde{\mathbf{m}}_i)^T \Sigma_i^{-1}(\mathbf{x} - \tilde{\mathbf{m}}_i)$$

where $\tilde{\mathbf{m}}_i$ and Σ_i are the estimation of the mean and total scatter matrix of class ω_i.

4 RESULTS

We have tested our method with 373 new samples from the CMU-Pittsburgh database [6], which do not belong to the learning set. We first project the samples into the Sorted PCA + LDA subspace, before recognition of the facial expressions using single classifiers or their combination: the results are reported in Table 1. The tested classifiers are: Mahalanobis distance (C_1), Euclidean distance (C_2), Bayes rules (C_3) and their combination (C_4). We observe that the combination of the three classifiers improves the correct classification rates: we then take into account all the properties and capacities of correct classification of single classifiers.

	Sur.	Ang.	Sad.	Hap.	Fea.	Dis.	**Tot.**
#	81	41	80	83	53	35	**373**
C_1	98%	91%	93%	86%	82%	73%	**87.2%**
C_2	98%	81%	95%	89%	73%	73%	**84.8%**
C_3	98%	95%	97%	91%	76%	73%	**88.3%**
C_4	98%	97%	97%	93%	84%	73%	**90.3%**

Table 1: Comparison of correct classification rates, after the projection of data onto Sorted PCA + LDA subspace, depending on the classifier: C_1 Mahalanobis distance, C_2 Euclidean distance, C_3 Bayes rule and C_4 fuzzy integral method.

5 CONCLUSION

We have presented a solution for the facial expression recognition problem, based on the combination of different classifiers using a fuzzy integral method. Results show that combining different classifiers can improve the classification accuracy. Such method allows to cover up some classifier failures and then to improve the classification accuracy. However, we have to find a compromise between the number of classifiers to combine and the global quality of classification: this method is more difficult to implement if we combine a lot classifiers, but it provides smaller classification error rates.

References

[1] J. N. Bassili. Emotion recognition: the role of facial movement and the relative importance of upper and lower areas of the face. *J. Personality and Social Psychology*, 37:2049–2059, 1979.

[2] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. *IEEE: Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, jul 1997.

[3] G. Donato, M.S. Bartlett, J.C. Hager, P. Ekman, and T.J. Sejnowski. Classifying facial actions. *IEEE: Transactions on Pattern Analysis and Machine Intelligence*, 21(10):974–989, oct 1999.

[4] S. Dubuisson, F. Davoine, and M. Masson. A solution for facial expression representation and recognition. In *ICAV3D*, Mykonos, Grece, may 2001.

[5] P. Ekman and W. Friesen. *Facial Action Coding System: a technique for the measurement of facial movements.* Calif.: Consulting Psychologists Press, 1978.

[6] T. Kanade, J. F. Cohn, and Y. Tian. Comprehensive database for facial expression analysis. In *Proceeding of the Fourth International Conference of Face and Gesture Recognition*, pages 46–53, Grenoble, France, 2000.

[7] J.M. Keller, P. Gader, H. Tahani, J.-H. Chiang, and M. Mohamed. Advances in fuzzy integration for pattern recognition. *Fuzzy sets and systems*, 65:273–283, 1994.

[8] L.I. Kuncheva, J.C. Bezdek, and R.P.W. Duin. Decision templates for multiple classifier fusion: an experimental comparison. *Pattern Recognition*, 34:299–314, 2001.

[9] M.J. Lyons, J. Budynek, and S. Akamatsu. Automatic classification of single facial images. *IEEE: Transactions on Pattern Analysis and Machine Intelligence*, 21(12):1357–1362, dec 1999.

[10] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[11] A. Webb. *Statistical pattern recognition.* Arnold, 1999.

[12] Y. Yacoob and L.S. Davis. Recognizing human facial expressions from long image sequences using optical flow. *IEEE: Transactions on Pattern Analysis and Machine Intelligence*, 18(6):636–642, jun 1996.